Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently Amended) A method of monitoring a computer application executed on a computer system, said method comprising the steps of:

without predefining events describing the potential stages of a transaction to be executed by said computer application, using an application program interface to add ~~adding~~ software code to said computer application for assigning a single general reference to characteristic transactional information associated with said transaction;

using said single general reference to identify transaction events performed by said computer application in executing said transaction; and

measuring ~~said transactions~~ desired transaction events.

2.   (Original) The method of claim 1 further comprising assigning a component-specific reference to said single general reference at each component of said computer system, said component-specific reference representing said characteristic transactional information as said computer application is executed on said computer system.

3. (Original) The method of claim 1 wherein said step of measuring comprises measuring the processing time spent by said computer application at each component of said computer system and measuring the processing time spent by said computer application between each component of said computer system.

4. (Original) The method of claim 3 further comprising the step of charting latency of said computer system over a selected time frame.

5. (Original) The method of claim 1 further comprising calculating latency of transactional information passed between components of said computer system according to the formula:

$$\frac{T'_1(U_{cy}) - T'_1(V_{cx})) + (T'_2(U_{cy}) - T'_2(V_{cx})) + \ldots + (T'_{m-1}(U_{cy}) - T'_{m-1}(V_{cx})) + (T'_m(U_{cy}) - T'_m(V_{cx}))}{m}$$

where:

m = an unspecified number of transaction events,

$T_1, T_2, \ldots, T_{m-1}, T_m$;

$T'_1, T'_2, \ldots, T'_{m-1}, T'_m$ = transactional information pertaining to transaction events, $T_1, T_2, \ldots, T_{m-1}, T_m$;

U_{cy} = start time for a transaction event at one component of said computer system; and

V_{cx} = end time for a transaction event at another component of said computer system.

6.   (Currently Amended) An application program interface for use in monitoring a computer application executed on a computer system comprised of at least one component, said application program interface comprising:

software code added to said computer application for assigning, without predefining events describing ~~the~~ potential stages of a transaction to be executed by said computer application, a single general reference to characteristic transactional information associated with a transaction to be executed by said computer application; and

an agent for marking ~~the~~ a time at which said software code is executed and tagging that time with said characteristic transactional information as said characteristic transactional information is being currently processed by the computer application.

7.   (Original) The application program interface of claim 6 wherein said software code is further operable to assign a component-specific reference to said single general reference at each component of said computer system, said component-specific reference representing said characteristic transactional information as said computer application is executed on said computer system.

8.   (Original) The application program interface of claim 6 wherein said agent measures the processing time spent by said computer application at each component of said computer system

and measures the processing time spent by said computer application between each component of said computer system.

9. (Currently Amended) A computer system performance monitoring system comprising:

an application program interface for monitoring a computer application executed on a computer system, said application program interface comprising:

software code added to said computer application for assigning, without predefining events describing ~~the~~ potential stages of a transaction to be executed by said computer application, a single general reference to characteristic transactional information associated with a transaction to be executed by said computer application; and

an agent for marking ~~the~~ a time at which said software code is executed and tagging that time with said characteristic transactional information as said characteristic transactional information is being currently processed by the computer application;

an aggregator for calculating computer application latency data from raw timing data produced by said agent; and

a database for storing said raw computer application timing data and said latency data.

10. (Original) The computer system performance monitoring system of claim 9 further comprising a graphical user interface for enabling a user to retrieve said latency data from said database.

11. (Original) The computer system performance monitoring system of claim 10 wherein said graphical user interface includes means for charting latency of a computer system over a selected time frame.

12. (Original) The computer system performance monitoring system of claim 9 wherein said aggregator calculates latency of transactional information passed between components of said computer system according to the formula:

$$\frac{T'_1(U_{cy}) - T'_1(V_{cx})) + (T'_2(U_{cy}) - T'_2(V_{cx})) + \ldots + (T'_{m-1}(U_{cy}) - T'_{m-1}(V_{cx})) + (T'_m(U_{cy}) - T'_m(V_{cx}))}{m}$$

where:

m = an unspecified number of transaction events, $T_1, T_2, \ldots, T_{m-1}, T_m$;

$T'_1, T'_2, \ldots, T'_{m-1}, T'_m$ = transactional information pertaining to transaction events, $T_1, T_2, \ldots, T_{m-1}, T_m$;

U_{cy} = start time for a transaction event at one component of said computer system; and

V_{cx} = end time for a transaction event at another component of said computer system.

7